

WHAT IS CLAIMED IS:

- 1 1. A method of synchronizing processors in a simulated
2 multiprocessor environment operable to execute a code portion
3 to be debugged, comprising the steps:
4 providing a synchronous breakpoint at a
5 predetermined address location with respect to said code
6 portion;
7 executing said code portion on said processors in
8 said simulated multiprocessor environment from a fixed
9 location; and
10 when a first processor of said processors
11 encounters said synchronous breakpoint, terminating execution
12 of said code portion on said first processor while continuing
13 to execute said code portion on remaining processors.

09843266 042504
#05240 5924860

1 2. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 1, further comprising the steps:

4 when a particular processor of said remaining
5 processors in said simulated multiprocessor environment
6 reaches said synchronous breakpoint, terminating execution
7 of said code portion on said particular processor while
8 continuing to execute said code portion in said simulated
9 multiprocessor environment until each of said remaining
10 processors has reached said synchronous breakpoint; and
11 thereafter, returning run control to a user
12 associated with said simulated multiprocessor environment.

1 3. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 1, wherein said step of providing a synchronous breakpoint
4 comprises converting a standard breakpoint into said
5 synchronous breakpoint.

1 4. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 3, wherein said standard breakpoint is converted into said
4 synchronous breakpoint by issuing a BREAKPOINT SYNC_SET
5 command.

1 5. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 4, further comprising the step of issuing a BREAKPOINT
4 SYNC_RELEASE command in order to release said processors from
5 said synchronous breakpoint.

1 6. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 4, wherein said step of executing said code portion on said
4 processors is performed in a round robin fashion with respect
5 to said processors.

1 7. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 4, wherein said code portion is selected from the group
4 consisting of an application program, a firmware code
5 portion, a booting sequence, a software tool, and an
6 operating system, and further wherein said synchronous
7 breakpoint comprises a breakpoint converted from a standard
8 breakpoint.

09843265 "04.25.01

1 8. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 1, wherein a debugger program is operated by said user
4 associated with said simulated multiprocessor environment.

1 9. The method of synchronizing processors in a
2 simulated multiprocessor environment as set forth in claim
3 8, wherein said debugger program is integrated with said
4 simulated multiprocessor environment.

044365 042504

1 10. A method of debugging a code portion targeted for
2 execution on a target hardware platform, comprising the steps
3 of:

4 providing an architectural simulator operable to
5 simulate said target hardware platform, said architectural
6 simulator having a debugger associated therewith, wherein
7 said target hardware platform comprises a multiprocessor
8 system;

9 initializing in said architectural simulator a list
10 of processors included in said target hardware platform;

11 setting a synchronous breakpoint at a predetermined
12 address location with respect to said code portion operable
13 to be executed on said architectural simulator;

14 launching said code portion on said architectural
15 simulator from a fixed location;

16 automatically stepping through said list of
17 processors until each of said processors of said
18 architectural simulator reaches said synchronous breakpoint;
19 and

20 thereafter, returning program control to said
21 debugger for performing a debug operation.

1 11. The method of debugging a code portion targeted for
2 execution on a target hardware platform as set forth in claim
3 10, wherein said multiprocessor system comprises a
4 symmetrical multiprocessor system.

1 12. The method of debugging a code portion targeted for
2 execution on a target hardware platform as set forth in claim
3 10, wherein said multiprocessor system comprises an
4 asymmetrical multiprocessor system.

1 13. The method of debugging a code portion targeted for
2 execution on a target hardware platform as set forth in claim
3 10, wherein said multiprocessor system comprises a loosely
4 coupled multiprocessor system.

1 14. The method of debugging a code portion targeted for
2 execution on a target hardware platform as set forth in claim
3 10, wherein said multiprocessor system comprises a tightly
4 coupled multiprocessor system.

1 15. The method of debugging a code portion targeted for
2 execution on a target hardware platform as set forth in claim
3 10, wherein said step of setting a synchronous breakpoint at
4 a predetermined address location is accomplished by
5 converting a standard breakpoint into said synchronous
6 breakpoint upon issuing a BREAKPOINT SYNC_SET command.

09843265 042407

1 16. A system for debugging a code portion targeted for
2 execution on a target hardware platform, comprising:
3 an architectural simulator operable to simulate
4 said target hardware platform, said architectural simulator
5 having a debugger associated therewith, wherein said target
6 hardware platform comprises a multiprocessor system;
7 means for initializing in said architectural
8 simulator a list of processors included in said target
9 hardware platform;
10 means for setting a synchronous breakpoint at a
11 predetermined address location with respect to said code
12 portion operable to be executed on said architectural
13 simulator;
14 means for launching said code portion on said
15 architectural simulator from a fixed location;
16 means for automatically stepping through said list
17 of processors until each of said processors of said
18 architectural simulator reaches said synchronous breakpoint;
19 and
20 means for returning program control to said
21 debugger for performing a debug operation after said
22 processors have reached said synchronous breakpoint.

1 17. The system for debugging a code portion targeted
2 for execution on a target hardware platform as set forth in
3 claim 16, wherein said multiprocessor system comprises a
4 symmetrical multiprocessor system.

1 18. The system for debugging a code portion targeted
2 for execution on a target hardware platform as set forth in
3 claim 16, wherein said multiprocessor system comprises an
4 asymmetrical multiprocessor system.

1 19. The system for debugging a code portion targeted
2 for execution on a target hardware platform as set forth in
3 claim 16, wherein said multiprocessor system comprises a
4 loosely coupled multiprocessor system.

1 20. The system for debugging a code portion targeted
2 for execution on a target hardware platform as set forth in
3 claim 16, wherein said multiprocessor system comprises a
4 tightly coupled multiprocessor system.

1 21. The system for debugging a code portion targeted
2 for execution on a target hardware platform as set forth in
3 claim 16, wherein said means for setting a synchronous
4 breakpoint at a predetermined address location is operable
5 to convert a standard breakpoint into said synchronous
6 breakpoint by issuing a BREAKPOINT SYNC_SET command.